2.4 SIGNATURE FLUCTUATIONS

The echo signal from a complex target in motion is rarely, if ever, constant. The variation in the echo signal may be caused by meteorological conditions, the lobe structure of the antenna pattern, equipment instabilities, or variations in the target cross section. The cross sections of complex targets (the typical target of interest to radars) are sensitive to aspect. Therefore, as the target aspect changes even slightly relative to the radar, variations in the echo signal will result. Fluctuations of the received target signal impact the predictions of both target detection and target tracking performance.

Although radar detection thresholds typically are calculated as the signal-to-noise ratio required to detect a single pulse, actual target detection occurs as the group of pulses in the observation period is integrated in some manner. Variation in the received target signal during the observation period causes a degradation in this integrated signal, resulting in an increase in the threshold required for target detection. Similarly, the improvement of the received signal-to-noise ratio (S/N) due to signal integration is degraded by signal fluctuations during the integration time. The large amplitude fluctuation of the radar cross section (RCS) with respect to small changes in the viewing angle is referred to as scintillation.

For target detection, proper accounting for target cross-section fluctuations involves use of the probability density function and the correlation properties with respect to time for a particular target and type of trajectory. The procedure to experimentally determine the correct density function and autocorrelation function requires an immense amount of data for each target and radar type. Usually, this is not practical. An economical method to assess the effects of a fluctuating cross section is to postulate a reasonable model for the fluctuations and to analyze it mathematically.

Several types of probability distributions have been proposed as reasonable models for signal fluctuations. Among these are the chi-square family, the log-normal family, the Swerling models, the Weinstock models, and the non-fluctuating model. Most of these models have been shown to match (approximately) some empirical data sets, but no general theory of target modulation exists. A subset of the Swerling models is simulated in *RADGUNS*. The Swerling models are described briefly in the following paragraphs.

The non-fluctuating case models a perfectly steady target echo. This is not realistic for real radar targets except in special cases such as spheres or targets which are stationary over the observation time. However, this model could be used to give detection estimates when minimal target information is available.

Swerling case 1 models a complex target consisting of many independent scatterers of approximately equal echoing areas. This model assumes that the echo pulses received from a target on any one scan are of constant amplitude throughout the entire scan but are independent (uncorrelated) from scan to scan. This assumption ignores the effect of the antenna beam shape on the echo amplitude. An echo fluctuation of this type is referred to as scan-to-scan fluctuation. Swerling case 2 models the same type of target as case 1. However, case 2 assumes that the fluctuations are more rapid than in case 1 and are independent from pulse to pulse.

Swerling case 3 models a target that can be represented as one large reflector combined with other smaller reflectors. The fluctuations are assumed to be independent from scan to scan as in case 1. Swerling case 4 models the same type of target as case 3, but with more rapid fluctuations that are independent from pulse to pulse (as in case 2).

The Swerling fluctuation models are special cases of chi-square distributions. Type 1 is of degree 2 and is referred to as the Rayleigh-power or exponential distribution.

Signal fluctuations also impact target angle and range tracking. A complex target can be envisioned as a set of scatterers with an apparent centroid that is a function of the relative locations and efficiency of each scatterer. Signal fluctuation causes movement of this apparent centroid, resulting in movement of the target tracking point. This phenomenon is known as glint or bright spot wander. Random distributions around the target centroid have been proposed as reasonable methods of describing glint. *RADGUNS* utilizes a sinusoidal variation about the target centroid to emulate glint.

Few, if any, real targets precisely fit any of the models mentioned above. Even if the exact statistical distribution of a target were known, the actual radar measurement of a target on a particular flight path might not clearly relate to that distribution. The Signal Fluctuations Functional Element is intended to generate changes in target signal returns that generally are accepted as realistic, and to simulate the effects of these changes on radar detection and tracking.

2.4.1 Functional Element Design Requirements

This section contains the design requirements to implement the signature fluctuations simulation in *RADGUNS*.

a. The fluctuations functional element will simulate the Swerling Type 1 fluctuating target model for radar detection.

b. For the tracking mode of radar operation, target glint will be modeled by a sinusoidal variation to the *x*, *y*, and *z* components of the target flight path. The horizontal and vertical components of the flight path will use bottom-view and front-view RCS values, respectively.

2.4.2 Functional Element Design Approach

This section describes the design approach that satisfies the design requirements of the previous section.

The probability of target detection depends on the signal-to-noise ratio (S/N) of received signals and the bias level of the radar system. The S/N is the ratio of the target signal to the sum of clutter return and thermal noise; fluctuations in the target signal received at the radar is modeled in this functional element.

The bias is the minimum radar detector voltage level that produces a target detection. The function describing bias depends on the combined effects of system noise and the integrator which combines several returned pulses for a resultant detection probability; these effects are out of scope of the Fluctuations FE. The description of bias level calculation is included

in the Threshold FE (Section 6.1). The description of (multiple radar pulses) integration is included in the Integration FE (Section 6.3).

Fluctuation effects are implemented by the form of the equation used to calculate probability of detection. The incomplete gamma function is a key component in probability of detection calculations. This function is described next, followed by explanation of probability of detection calculations.

Design Element 4-1: Incomplete Gamma Function

The Fluctuations FE uses the incomplete gamma function in the probability of detection model. A modified version of this function defined by Pearson is used in *RADGUNS*; its definition is given as Equation [2.4-3].

For an integer P, the incomplete gamma function is defined as (Reference 9, equation A-1)

$$f(a,P) = \int_0^a \frac{e^{-u}u^P}{P!}\,du \qquad [2.4\text{-}1]$$

An alternate form is (Reference 9, Equation A-2)

$$f(a,P) = 1 - \int_a^\infty \frac{e^{-u}u^P}{P!}\,du \qquad [2.4\text{-}2]$$

The incomplete gamma function can be modified by the substitution $a = u(P+1)^{1/2}$ to obtain Pearson's form of the incomplete gamma function, I(u,P) (Reference 9, Equation 2-16):

$$I(u,P) = \int_0^{u\sqrt{P+1}} \frac{e^{-\delta}\delta^P}{P!}\,d\delta \qquad [2.4\text{-}3]$$

Design Element 4-2: Approximation of the Incomplete Gamma Function

The incomplete gamma function is solved numerically in series form. These series are derived through repeated integration by parts of Equation [2.4-1] (Reference 9, Section A-2). For f(a,P), two approximations are used; the choice depends on the arguments a and P. If a ≥ P, Equation [2.4-1] converges rapidly, and the following series solution for the incomplete gamma function is used (Reference 9, Equation A-5):

$$f(a,P) = 1 - \sum_{k=0}^{P} \frac{e^{-a}a^k}{k!}, \quad a \geq P \qquad [2.4\text{-}4]$$

For a < P, Equation [2.4-2] converges rapidly, and its series representation is as follows (Reference 9, Equation A-7):

$$f(a,P) = \sum_{k=P+1}^{\infty} \frac{e^{-a}a^k}{k!}, \quad a < P \qquad [2.4\text{-}5]$$

An alternate representation of the terms of Equation [2.4-5] is derived as follows:

$$\begin{aligned}\frac{e^{-a}a^k}{k!} &= \frac{e^{-a}e^{[\ln(a^k)]}}{e^{[\ln(k!)]}} \\ &= e^{[k\ln(a)-\ln(k!)]} \\ &= e^{\left[-a+k\ln(a)-\sum_{t=1}^{k}\ln(i)\right]}\end{aligned} \qquad [2.4\text{-}6]$$

Since ln(1) = 0, the $i = 1$ term of the summation can be dropped to yield a final form used in *RADGUNS*:

$$\frac{e^{-a}a^k}{k!} = e^{\left[-a+k\ln(a)-\sum_{i=2}^{k}\ln(i)\right]} \qquad [2.4\text{-}7]$$

Design Element 4-3: Probability of Detection Model

Many distribution functions have been used to describe target fluctuations. In *RADGUNS*, the user selects one of two target detection models. Only one of these, the probability of detection model, incorporates target fluctuations. The variables effecting probability of detection calculations are S/N, probability of false alarm (P_{fa}), and number of pulses integrated (N). The calculated probability is compared with a user-defined value for detection. Calculation of bias, S/N, N, and the actual comparison are included in other FEs. The fluctuations FE implements fluctuations in terms of probability of target detection by use of the Swerling type 1 equations. The probability of detection, P_d, is given by the following expressions that depend on N, the number of pulses integrated.

For $N = 1$ (Reference 9, Equation 3-52),

$$P_d = e^{-\left(\frac{Y_b}{1+\bar{x}}\right)} \qquad [2.4\text{-}8]$$

where: $\bar{x}$ = average signal-to-noise ratio
Y_b = bias level

For $N > 1$ (Reference 9, Equation 3-56),

$$\begin{aligned}P_d = 1 - I\left[\frac{Y_b}{\sqrt{(N-1)}},\ N-2\right]\left(1+\frac{1}{N\bar{x}}\right)^{N-1} \\ \cdot I\left[\frac{Y_b}{\left(1+\frac{1}{N\bar{x}}\right)\sqrt{(N-1)}},\ N-2\right]e^{-\left(\frac{Y_b}{1+N\bar{x}}\right)}\end{aligned} \qquad [2.4\text{-}9]$$

where: $I[u,P]$ = Pearson's form of the incomplete gamma function (Reference 9, Equation 2-16)

The probability of detection is used for comparison with a user-defined value for target detection. Signal Processing Threshold FE encompasses detection status considerations.

Design Element 4-4: Approximation of the Probability of Detection Model

RADGUNS used Pearson's form of the incomplete gamma function in calculating the probability of detection for a Swerling case 1 model. By using Equation [2.4-5] above as an approximation to the incomplete gamma function, the approximation to the probability of detection, Equation [2.4-9], can be expressed as a series solution (Reference 9, Section A-4):

$$P_D = 1 - \sum_{k=N-1}^{\infty} \frac{exp(-Y_b)}{k}(Y_b)^k + \left(1 + \frac{1}{N\bar{x}}\right)^{N-1} \sum_{k=N-1}^{\infty} \frac{exp\left[\frac{-Y_b}{1+(1/Nx)}\right]}{k!} \cdot \left[\frac{Y_b}{1+(1/N\bar{x})}\right]^k exp\left[\frac{-Y_b}{1+N\bar{x}}\right] \qquad [2.4\text{-}10]$$

by letting for the first summation

$$a = Y_b$$

$$and \qquad [2.4\text{-}10a]$$

$$P = N - 2$$

and, similarly, for the second summation

$$a = \frac{Y_b}{1+(1/N\bar{x})}$$

$$and \qquad [2.4\text{-}10b]$$

$$P = N - 2$$

where: Y_b = bias level
$\bar{x}$ = average signal-to-noise ratio
N = number of pulses integrated

Design Element 4-5: Fluctuations Due to Glint

Changes in the target aspect with respect to the radar can cause the apparent center of radar reflections to wander from one point to another. The random wandering of the center of reflection produces angle tracking errors, and this type of error or noise is called target glint (Reference 1, p. 168). The wandering of the center of reflection is simulated in *RADGUNS*

by adding a sinusoidal variation to the target flight path; this artificially induces tracking errors since the target position is changed with respect to the track coordinates.

The target sinusoidal excursion, E_i, from the flight path is governed by a sine function for each axis ($i = x, y,$ or z) of the scenario coordinate system:

$$E_i = G_i \sin \omega t \qquad [2.4\text{-}11]$$

where: G_i = amplitude calculated in Equation [2.4-12]
ω = angular frequency of fluctuations
t = time since initiation of tracking (s)

The amplitude of the flight path sinusoidal variation along the coordinate axis i, G_i, is calculated as the radius of a circle of area equivalent to the appropriate target presented area, TPA_i:

$$G_i = \left(\frac{TPA_i}{\pi}\right)^{\frac{1}{2}} \qquad [2.4\text{-}12]$$

The TPA data value used for the x and y axes, TPA_x and TPA_y, is that of the bottom-view of the target. The front-view TPA is used for the z axis.

RADGUNS modifies Equation [2.4-11] to use the slant range, R, between the radar and the target, instead of ωt. The argument of the sine function, as implemented, is derived by finding a value Q so that

$$\frac{R}{Q} = \frac{Vt}{Q} = 2\pi ft = \omega t \qquad [2.4\text{-}13]$$

where: Q = adjustment factor
V = target speed (m/s)
t = time (s)
f = frequency of fluctuations (Hz)
ω = angular velocity (rad/s)

Thus, Equation [2.4-11] is implemented in *RADGUNS* as

$$E_i = G_i \sin\left(\frac{R}{Q}\right) \qquad [2.4\text{-}14]$$

Equation [2.4-13] can be solved for Q if V and f are known. Glint has a frequency range of fluctuations, f, between 4 Hz and 10 Hz. (Reference 10, page 230). The model developer assumed a 10 Hz frequency for fluctuations due to glint. The model developer assumed a nominal target speed V (for a typical engagement) of 200 m/s.

Solving Equation [2.4-13] for Q gives $Q = 3.183$. The developer chose slight variations in the Q to introduce a phase shift in the x, y, and z components (Q_x, Q_y, Q_z) of the sinusoidal variation; these are as follows:

Q_i = 3.0 for x-axis contribution
= 3.1 for y-axis contribution
= 3.2 for z-axis contribution

2.4.3 Functional Element Software Design

This section contains the software design necessary to implement the functional element requirements and the design approach outlined above. It is organized as follows: the first subsection describes the subroutine hierarchy and gives descriptions of the relevant subroutines; the next subsection contains a logical flow chart and describes all important operations represented by each block in the chart; the last subsection contains a description of all input and output data for the functional element as a whole and for each subroutine which implements fluctuations.

Subroutine Design

Figure 2.4-1 shows the calling sequence of the fluctuations functional element within the entire model structure. Functions which implement the FE are in shaded blocks. Subroutines which use the functional element's output appear in plain blocks. Each of these subroutines is described briefly in Table 2.4-1.

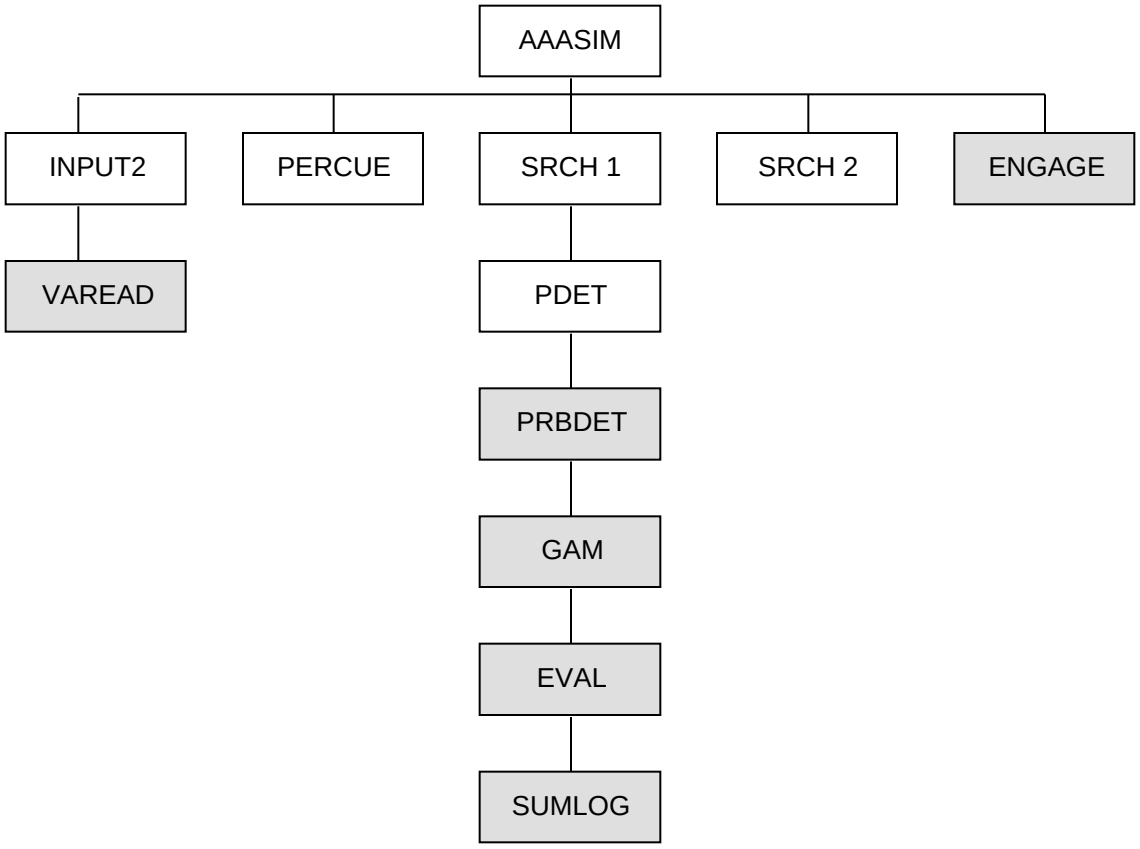

FIGURE 2.4-1. Subroutine Call Tree.

TABLE 2.4-1. Module Description.

Module Name	Description
AAASIM	Main routine to simulate AAA system
PERCUE	Searches for target with antenna perfectly cued to target's position
SRCH1	Searches for target in sector search or slow circular scan mode
SRCH2	Searches for target in circular scan mode
ENGAGE	Controls AAA system while in autotrack mode
PDET	Calculates number of pulses integrated
PRBDET	Computes Marcum-Swerling detection probabilities
GAM (B,N,TN)	Function, calculates $1 - \sum_{j=0}^{N} e^{-B+j\ln(B)-\sum_{i=2}^{N}\ln(i)}$

TABLE 2.4-1. Module Description. (Contd.)

Module Name	Description
EVAL (Y,N)	Function, calculates $e^{-Y+N\ln(Y)-\sum_{i=2}^{N}\ln(i)}$
SUMLOG (N)	Function, calculates $\sum_{i=2}^{N}\ln(i)$
INPUT2	Controls data initialization
VAREAD	Reads target vulnerable and presented area data

Note: Modules implementing the fluctuations functional element are identified in shaded cells.

Logical Flow for Subroutine PRBDET. The primary routine which implements fluctuations modeling is subroutine PRBDET. PRBDET relies on functions GAM, EVAL, and SUMLOG. The diagram in Figure 2.4-2 shows the logical flow of computations related to the FE. Variable names are enclosed in parentheses.

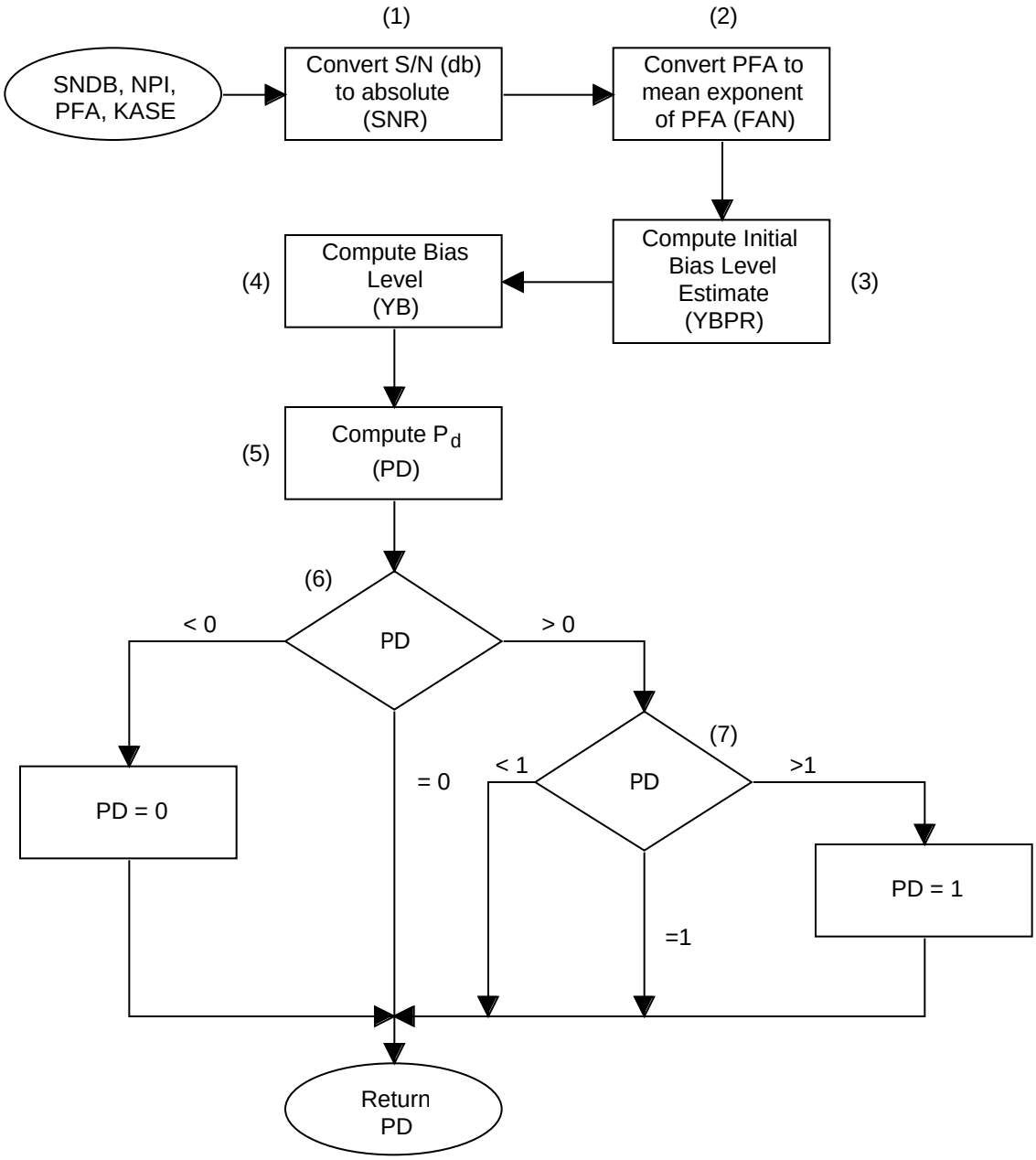

FIGURE 2.4-2. PRBDET Subroutine Flow Chart.

Block 1. The S/N is passed (as an argument) to PRBDET as variable SNDB. This variable is converted from decibels to an absolute ratio and is stored in variable SNR.

Block 2. The false alarm number is passed as an argument to PRBDET as variable PFA. The false alarm number is defined as the absolute value of the exponent of the power of ten in the probability of false alarm expression (e.g. PFA = 10 means FAN = 8). This number is set to the constant value of 8 in the calling routine (PDET). It is used indirectly by fluctuations since it is incorporated in the bias level calculation; bias also is an input used by the Fluctuations FE.

Block 3. An initial estimate of the bias level is computed and stored in variable YBPR. This initial value, and the final value determined numerically in Block 4 are described in the Threshold FE.

Block 4. The final value of the bias level (variable YB) is found numerically, and is used in probability calculations described in Block 5.

Block 5. The probability of detection, P_d, is calculated for a Swerling 1 target and is stored in variable PD. The calculations are described in Section 1.2.2.3.3. P_d is dependent on the number of pulses integrated, which is passed as an argument (variable NPI). P_d also depends on the average S/N (variable SNR). SNR is calculated from the S/N power ratio in decibels passed as an argument (SNDB) to PRBDET. The bias level, calculated in Block 4 (YB), and SNR are used in the exponential equation (Equation [2-8]) for the P_d for a single pulse. The same variables also are used in calculating PD for multiple pulses integrated as defined by Equation [2.4-9]; in this case, several terms are included in the calculation of PD. The implementation of PD for NPI>1 is based on Equation [2.4-10]. The approximation is accomplished through calls to Function GAM, which implements Pearson's form of the incomplete gamma function defined by Equation [2.4-3]. GAM is described in the next section.

Block 6 and 7. The algorithm in Block 5 does not limit PD to values between 0 and 1, so Blocks 6 and 7 set the probability of detection to 0 when PD < 0, and to 1 when PD > 1.

Logical Flow for Function GAM. GAM(B,N,TN) implements the series solutions of the incomplete gamma function. Function arguments B and N correspond to arguments a and P, respectively, in Equation [2.4-4]. The implementation of a and P is defined by Equations [2.4-10a and -10b]. The third argument of GAM, TN, is an output that is not utilized in calculations of target fluctuations.

Figure 2.4-3 depicts the functional flow of GAM. Two main execution paths are possible. One path is followed when B < N (Blocks 1 - 6); this path implements Equation [2.4-5]. The other path is executed when B complete gamma function converges more rapidly for this latter case by use of Equation [2.4-4]. The terms in Equation [2.4-4] are calculated using Equation [2.4-7]. A recursive loop is used to accumulate terms by calling Function EVAL.

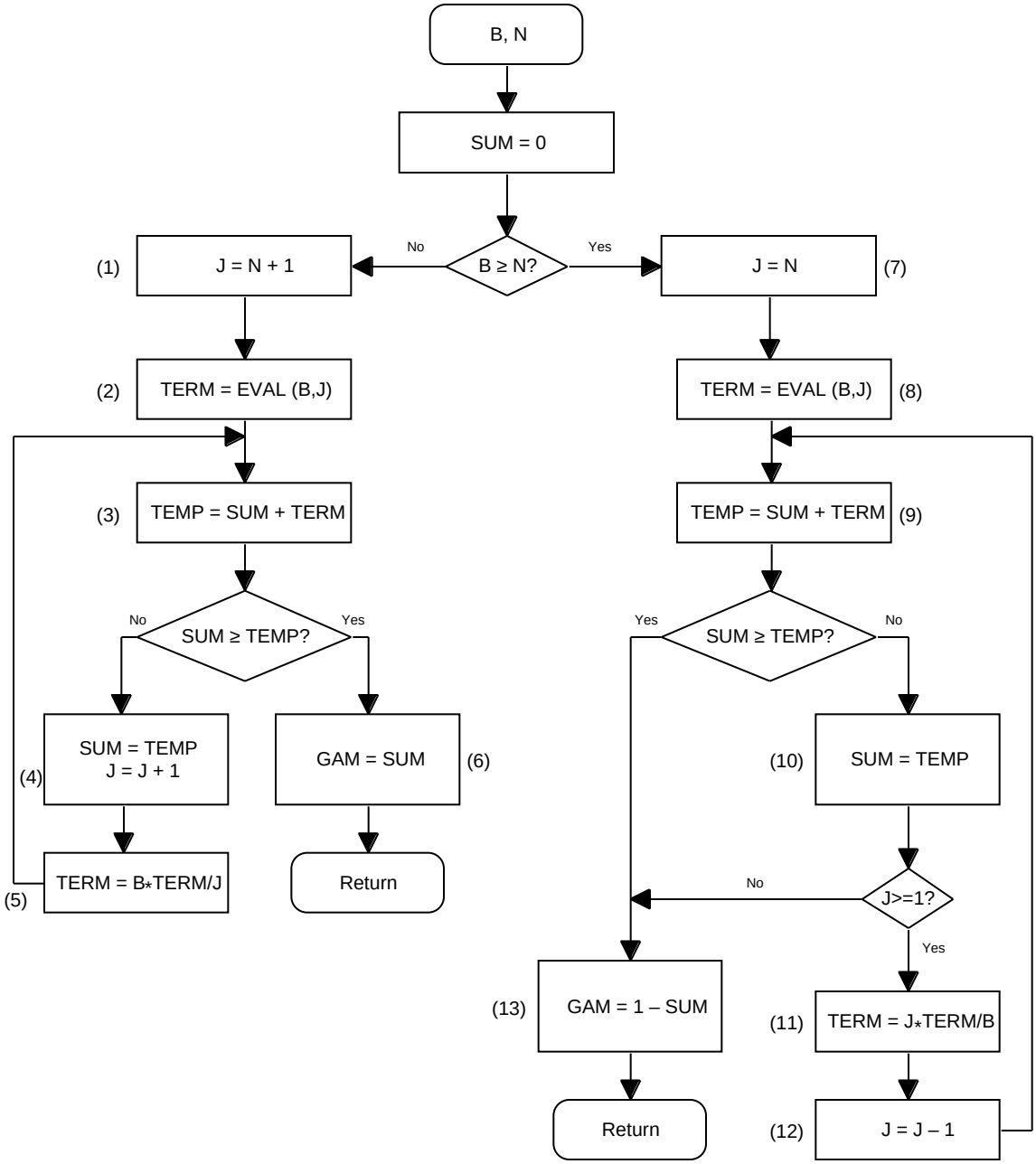

FIGURE 2.4-3. Function GAM Logic Flow.

For the condition B < N, Blocks 1 - 6 are the path of execution. These blocks are described next.

Blocks 1 - 2. Two local variables, J and TERM, are initialized. J is the lower bound of integration of the incomplete gamma function, and hence is the lower index for the series approximation of the function. The first term of the series also is initialized (variable TERM).

Blocks 3 - 6. Each successive term is calculated and accumulated in the loop represented by Blocks 3, 4, and 5. When a term becomes sufficiently small (as determined in Function EVAL), its value is set to zero. This condition causes execution to branch to Block 6, where the solution to the incomplete gamma function (the value of GAM) is returned to the calling routine.

For the condition B are described next.

Blocks 7 - 8. These are analogous to Blocks 1 and 2.

Blocks 9 - 13. These are analogous to Blocks 3-6. The major difference is that J is decremented rather than incremented, and thus an additional exit criterion exists. The loop is exited after processing J=1.

Logical Flow for Function EVAL. EVAL(Y,N) calculates each term of the series solution of the incomplete gamma function. Each term has a solution defined by Equation [2.4-7]. For this FE, the arguments Y and N are functions of the sensor bias level and the number of pulses integrated, respectively, and are defined by Equations [2.4-10a and -10b].

The final term (the summation) in the exponent of Equation [2.4-7] is calculated by a call to Function SUMLOG. SUMLOG is described in the next section. The logic flow of EVAL is depicted in Figure 2.4-4.

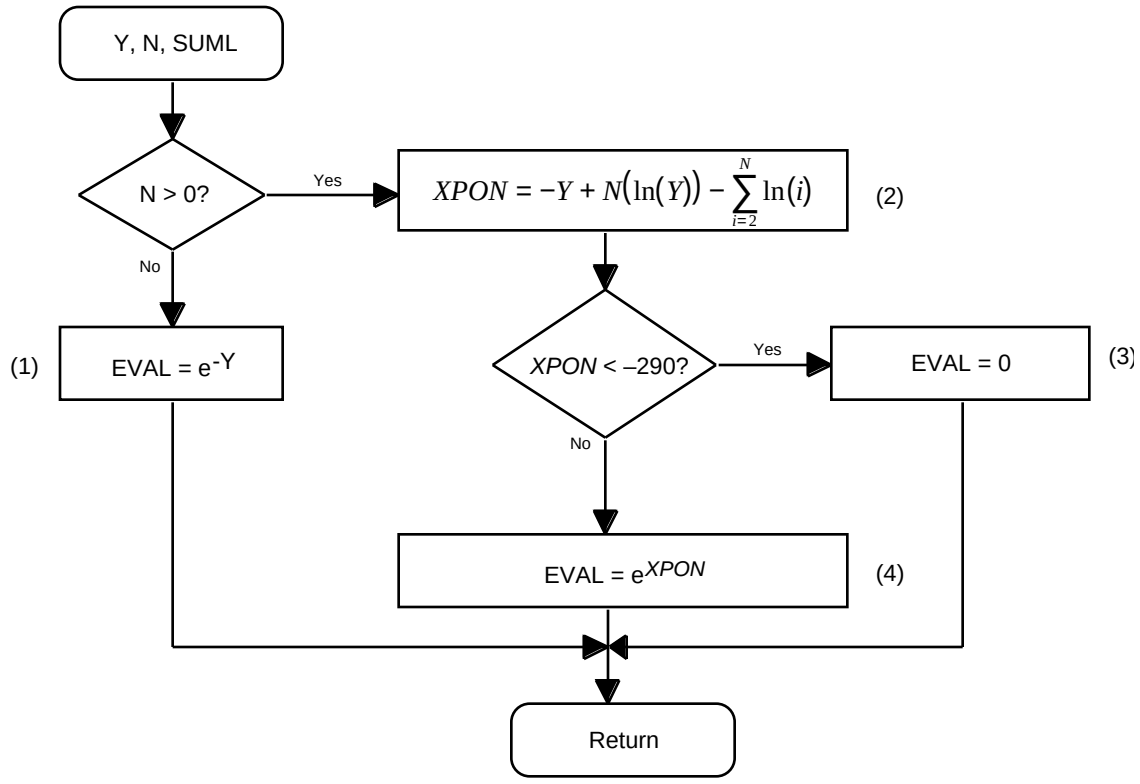

FIGURE 2.4-4. Function EVAL Flow Chart.

Two branches for calculating the function value are available depending on the value of N. If N < 0, Block 1 is performed. If N > 0 the branch encompassing Blocks 2 - 4 is performed. Block 1 gives the same result as would Blocks 2 - 4 if N=1, but is much more efficient computationally.

Block 1. If N < 0, Equation [2.4-7] reduces to a simplified form (e^Y), which is implemented in one step.

Block 2. If N > 0, the exponent of Equation [2.4-7] is calculated as variable XPON.

Block 3. For the case of XPON <.

Block 4. For the case of XPON ^{XPON}.

Logical Flow for Function SUMLOG. Function EVAL(Y,N) calls Function SUMLOG(N), where input argument N is the same for both functions (NPI-2 for the fluctuations FE). The final term in the exponent of Equation [2.4-7], the summation of the natural logarithms of integers between 2 and N, is calculated by SUMLOG.

Several local variables (DUMA, DUMB, NMAX) first are initialized. NMAX=1000 is an array size limit for variable A which accumulates the logarithms. This limit thus defines the maximum number of loops executed in several recursive algorithms in the module. Another set of initializations occurs as described in Block 1. Figure 2.4-5 is the functional flow diagram for SUMLOG.

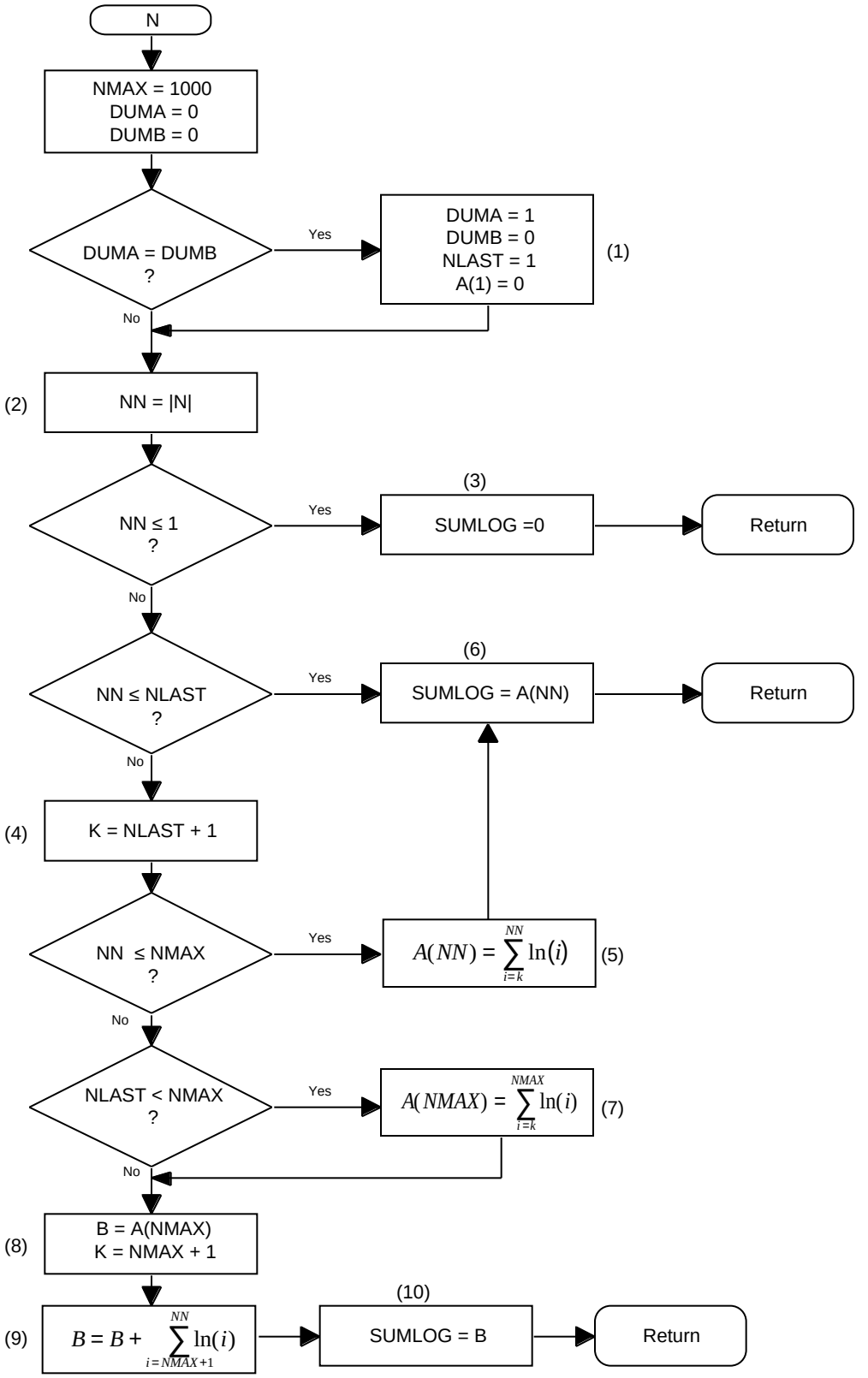

FIGURE 2.4-5. SUMLOG Logical Flow.

Block 1. The data initialization at the start of SUMLOG always results in the condition DUMA = DUMB. Thus, *RADGUNS* always executes Block 1 even though a conditional statement implies that these initializations could be bypassed.

Block 2. The local variable NN (actually a constant) is set to the absolute value of N (i.e., |N|). This step ensures that only positive integers (as well as zero) are used as inputs to the natural logarithm operation.

Block 3. If NN < 1, the value of the function returned to the calling routine is set to zero, and execution exits SUMLOG.

If NN > 1, a condition NN < NLAST implies that a direct branch to Block 6 could occur. However, NLAST = 1 and NN > 1, so Block 6 is never reached at this point.

Up to 1000 partial sums of the series of natural logarithms are stored in array A. If all 1000 array entries are used, execution branches to a summation routine (at Block 8) that continues to accumulate the natural logarithms of numbers greater than 1000. However, the accumulation is stored recursively in variable B, which does not use an array.

Block 4. If NN > 1, the final term in the summation has not been calculated. The variable K is set to NLAST+1 = 2. K is the starting point of do-loops which calculate the summations in several blocks which follow.

Block 5. If NN < NMAX, a recursive loop accumulates the natural logarithms of integers from K up to and including NN. The partial sums are stored in array A.

Block 6. The value of SUMLOG is set to the highest (NN^{th}) array value.

If NN > NMAX, execution branches to Blocks 7 - 10 after Block 4. A second conditional statement implies that Block 7 could be bypassed, but NLAST = 1 and NMAX = 1000, so Block 7 always is executed if NN > NMAX.

Block 7. If NN > NMAX, a recursive loop accumulates the natural logarithms of integers from K(=NLAST+1) to NMAX(=1000) in successive array elements of variable A.

Block 8. B, the variable in the next block which accumulates logarithms, is set to A(NMAX). K, the lower bound of the summation which continues the processes equivalent to those of Block 7, is set to NMAX + 1.

Block 9. The accumulation of natural logarithms continues for integers larger than 1000. A recursive loop accumulates in variable B the natural logarithms of integers up to and including NN, where NN is the absolute value of the argument (N) passed to SUMLOG. This essentially is the end of the subroutine computations for this path, since SUMLOG is set to B in the next step.

Block 10. The value of SUMLOG returned to the calling routine is set equal to B.

Logical Flow for the Implementation of Glint. The functional flow diagram for the implementation of glint is depicted in Figure 2.4-6. The amplitude of flight path sinusoidal variation to emulate glint is based on the target bottom- and front-view presented area

(TPA) data, if they exist. If they do not exist, default values are incorporated in calculations described in Block 1. The sinusoidal variation of the flight path is a function of the target range. VA data and target range are inputs to glint modeling, as shown in the top oval of the flow diagram.

The 6-view and 26-view vulnerable area data files are checked for the applicable VA data. The 6-view data are used if both files have data.

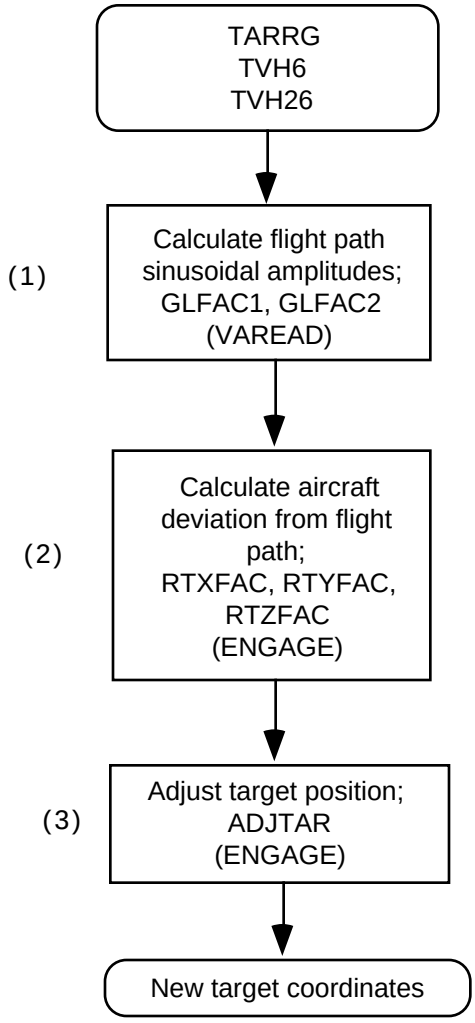

FIGURE 2.4-6. Glint Functional Flow Chart.

Block 1. The bottom-view TPA data are used to calculate amplitudes of both the *x*- and *y*-axis contributions; this value is stored as variable GLFAC1. The front-view TPA is used to calculate the *z*-axis amplitude which is stored as variable GLFAC2. These two variables are calculated in subroutine VAREAD as defined by Equation [2-11]. If neither have data, default values assigned to GLFAC1 and GLFAC2 are 5.0 and 1.0, respectively.

Block 2. Common Block GLINT is used to pass GLFAC1 and GLFAC2 to subroutine ENGAGE, where they are used as amplitudes of the sinusoidal variation to the target flight

path defined by Equation [2-12]; this equation is implemented as variables RTXFAC, RTYFAC, and RTZFAC in ENGAGE for the *x*, *y*, and *z* axes, respectively.

Block 3. RTXFAC, RTYFAC, and RTZFAC are added to the cartesian coordinates of the target for an adjusted target position. The adjusted target position (as well as the existing speed and acceleration) is stored in the array variable ADJTAR.

The offset from the original flight path is accounted for by the tracking system. The result is increased tracking errors, which is the intent of glint modeling in *RADGUNS*.

Functional Element Inputs And Outputs

This section identifies the input and output data associated with the FE. Table 2.4-2 shows the FE output.

TABLE 2.4-2. FE Outputs.

Variable Name	Description
PD	Probability of *detection* (in acquisition mode)
ADJTAR	Adjusted target position used to produce target *tracking* errors to emulate glint

Table 2.4-3 displays the user-defined input data to the FE. The detection type (PDET) is highlighted in bold characters because it must be selected by the user so that the Swerling Type 1 fluctuation model can be executed.

TABLE 2.4-3. User Inputs to FE.

Variable Name	Variable Options	Description
DETTYP	User Input (**PDET**, THRS)	Select either probability of detection model or threshold detection model

Some inputs to modules implementing the FE are used to calculate variables specific (local) to a module that also are inputs to the FE. These local variables are described in Table 2.4-4.

TABLE 2.4-4. Local Variable Inputs to FE.

Variable Name	Module	Description
YB	PRBDET	Sensor bias level
SNR	PRBDET	Absolute signal-to-noise power ratio

Subroutine PDET calculates the variables passed as arguments to PRBDET. The routine which primarily implements fluctuations is PRBDET. PRBDET inputs and outputs are defined in Table 2.4-5.

TABLE 2.4-5. Subroutine PRBDET Inputs and Outputs.

SUBROUTINE: PRBDET					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
SNDB	Argument	Signal-to-Noise power ratio in decibels	PD	Argument	Probability of detection
NPI	Argument	Number of pulses integrated			
PFA	Argument	Probability of false alarm			
KASE	Argument	Swerling Fluctuation Model			

Subroutine PRBDET uses several functions to calculate often-used mathematical formulas. Tables 2.4-6, 2.4-7, and 2.4-8 define the inputs and outputs of those functions.

TABLE 2.4-6. Function GAM Inputs and Outputs.

FUNCTION: GAM					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
B	Argument	Used in exponential series (see equation 2-10)	GAM	Function Value Returned	Numerical approximation to the incomplete gamma function
N	Argument	Used in exponential series (see equation 2-10)	TN	Argument	Not used by fluctuations FE

TABLE 2.4-7. Function EVAL Inputs and Outputs.

FUNCTION: EVAL					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
Y	Argument	Used in exponential equation (see equation 2-10)	EVAL	Functional Value Returned	Evaluation of a single term of the incomplete gamma function
N	Argument	Used in exponential equation (see equation 2-10)			

TABLE 2.4-8. Function SUMLOG Inputs and Outputs.

FUNCTION: SUMLOG					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
N	Argument	Used to define number of terms in summation	SUMLOG	Function Value Returned	Computes sum of natural logarithms of integers from 1 to N.

Fluctuations due to glint introduce tracking errors. The error is introduced by producing target excursions from the user-defined target flight path. The excursions are a function of target presented area (TPA) data read by subroutine VAREAD; Table 2.4-9 describes the inputs and outputs related to glint calculations. TPA data are used by subroutine ENGAGE to calculate new target positions; Table 2.4-10 describes the inputs and outputs of ENGAGE that are related to glint calculations.

TABLE 2.4-9. Subroutine VAREAD Inputs and Outputs.

SUBROUTINE: VAREAD					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
TVH6(1)	common	6-view presented area (TPA) data, front view	GLFAC1	common	Amplitude of horizontal sinusoidal flight path variation to emulate glint
TVH6(5)	common	6-view TPA data, bottom view	GLFAC2	common	Amplitude of vertical sinusoidal flight path variation to emulate glint
TVH26(1)	common	26-view TPA data, bottom view, lat/long = -90û/90û			
TVH26(14)	common	26-view TPA data, front view, lat/long = 0û/0û			

TABLE 2.4-10. Subroutine ENGAGE Inputs and Outputs.

SUBROUTINE: ENGAGE					
Inputs			**Outputs**		
Name	**Type**	**Description**	**Name**	**Type**	**Description**
GLFAC1	Common	Amplitude of horizontal sinusoidal flight path variation to emulate glint	ADJTAR	Common	Adjusted target position (to produce track errors)
GLFAC2	Common	Amplitude of vertical sinusoidal flight path variation to emulate glint			
TARRG	Output from call to POLAR	Slant range between target and radar			

2.4.4 Assumptions and Limitations

The fluctuations functional element is limited to modeling of the Swerling Type 1 target fluctuation for detection and to modeling of glint for tracking.

Glint modeling assumes a target speed of 200 m/s and a frequency of target signature fluctuation (due to glint) of 10 Hz.